Final PostgreSQL DBA Training Plan

Local PostgreSQL + RDS Concepts (No RDS Hands-On)

Duration: 4 Days × 4 Hours = 16 Hours

Audience: Senior DBA Professionals (10+ years, Oracle / SQL Server)

Organization Context: Fidelity International (Investment, Retirement, Financial Services)

Tools: psql, pgAdmin 4, Pagila DB

Assessment: ✅ 40-Question Post Test + Module-wise FAQ

Installation: ✅ Completed by participants **before training**

Pre-Training Requirement (Mandatory)

Participants must complete **before Day 1**:

- Install PostgreSQL (Windows/macOS)
- Verify psql and pgAdmin access
- Download Pagila dataset
- Confirm local connectivity

(No classroom time spent on installation)

DAY 1 — PostgreSQL Foundations, Architecture & Tools (4 Hours)

A. PostgreSQL Introduction — Condensed (30 minutes)

Topics

- PostgreSQL capabilities overview
- PostgreSQL vs Oracle vs SQL Server vs AWS RDS PostgreSQL
- Architecture differences (Local vs RDS)
- MVCC high-level concept
- Key differences vs enterprise databases

Fidelity Use Case

- PostgreSQL used for **investment reporting, portfolio views, microservices**
- Oracle retained for **core settlement / legacy systems**
- RDS PostgreSQL used for **managed production workloads**

B. PostgreSQL Local Architecture (1 hour)

- Cluster structure (data directory, instance, databases, schemas)
- Key processes (postmaster, autovacuum, WAL writer, checkpointer)
- Memory architecture
- WAL, checkpoints, background writer
- MVCC row versioning

Fidelity Use Case

- Gurugram supports multi-DB clusters → shared memory understanding critical
- UK latency SLAs → WAL & checkpoint behavior matters
- China read-heavy compliance queries → MVCC benefits

Break — 15 minutes

C. Tools Overview & Connectivity (30 minutes)

- psql vs pgAdmin (DBA usage patterns)
- Connecting locally
- Roles, databases, schemas
- Pagila dataset overview

D. Hands-On Lab (45 minutes)

- Connect using psql and pgAdmin
- Create roles, schemas, objects
- Load Pagila schema + data
- Run basic queries
- Measure performance using:
  - \timing
  - EXPLAIN / EXPLAIN ANALYZE
- Explore system catalogs:
  - pg_class
  - pg_stat_activity

Module-wise FAQ — Day 1

Q1. Is PostgreSQL production-ready for financial systems?

Yes. PostgreSQL is ACID-compliant and widely used in regulated financial environments.

Q2. How is PostgreSQL different from Oracle RAC?

PostgreSQL does not have shared-disk RAC. HA is achieved via replication and managed services like RDS Multi-AZ.

Q3. Why do DBAs still need architecture knowledge in RDS?

Because performance, MVCC, vacuum, and query plans are **not managed by AWS**.

Q&A — 30 minutes

DAY 2 — Transactions, Locking & Maintenance (4 Hours)

---

1. Transaction Deep Dive & Isolation Levels (1 hour)

- MVCC internals
- Snapshots & visibility rules
- Transaction IDs
- Isolation levels
- Comparison with Oracle & SQL Server

Fidelity Use Case

- Trading systems → Read Committed
- End-of-day reconciliation → Serializable

---

2. Locks, Blocking & Deadlocks (1 hour)

- Row & table locks
- Advisory locks
- Blocking chains
- Deadlock detection
- Index impact on locking

Fidelity Use Case

- Reporting queries blocking trade updates
- Advisory locks for batch reconciliation jobs

Break — 15 minutes

---

3. Vacuum & Autovacuum Internals (1 hour)

- Dead tuples & bloat
- VACUUM vs VACUUM FULL
- Autovacuum thresholds
- Visibility & Free Space Maps
- XID wraparound

Fidelity Use Case

- Order lifecycle tables with heavy UPDATEs
- RDS performance incidents caused by ignored autovacuum

---

4. Lab + Q&A (45 minutes)

- Generate dead tuples
- Run VACUUM / VACUUM FULL
- Compare table sizes
- Inspect autovacuum behavior

Module-wise FAQ — Day 2

Q1. Why does PostgreSQL need VACUUM when Oracle has undo?

PostgreSQL stores old row versions in the table itself, not in undo segments.

Q2. Is VACUUM FULL safe in production?

No. It requires exclusive locks. Use only during maintenance windows.

Q3. Does RDS handle vacuum automatically?

Autovacuum runs automatically, but **tuning and monitoring remain DBA responsibilities**.

DAY 3 — Query Optimization & Indexing (4 Hours)

---

1. Planner Internals (1 hr 15 min)

- Cost-based optimizer
- Scan types
- Join methods
- EXPLAIN vs EXPLAIN ANALYZE
- Statistics & planner accuracy

Break — 15 minutes

---

2. Indexing Strategies (1 hour)

- B-Tree, Hash, GIN, BRIN
- Partial indexes
- Multi-column indexes
- INCLUDE (covering indexes)
- Index-only scans

Fidelity Use Case

- Partial indexes for active trades
- BRIN for time-series transaction history

---

3. Advanced Performance Tuning (45 minutes)

- shared_buffers
- effective_cache_size
- work_mem
- maintenance_work_mem
- random_page_cost
- parallelism
- Query rewrites

---

4. Lab + Q&A (45 minutes)

- Tune slow Pagila queries
- Add indexes
- Compare plans before/after
- Measure improvements

Module-wise FAQ — Day 3

Q1. Why does PostgreSQL sometimes choose sequential scans?

Because the planner estimates it to be cheaper based on statistics.

Q2. Are more indexes always better?

No. Indexes slow down INSERT/UPDATE/DELETE operations.

Q3. Can we use hints like Oracle?

Not natively. PostgreSQL relies on statistics-driven planning.

DAY 4 — Backup, Replication Concepts & Migration (4 Hours)

---

1. Backup & Restore (1 hour)

- pg_dump / pg_restore
- Backup formats
- Schema-only restores
- Restore sequencing

RDS Concepts

- Automated backups
- Snapshots
- PITR
- Snapshot vs logical backup

---

2. Replication Concepts (1 hr 15 min)

(Conceptual only)

- Physical vs logical replication
- WAL shipping
- Replication slots

- Failover concepts (RPO / RTO)

Fidelity Use Case

- UK / India read replicas for reporting
- China data locality & compliance

Break — 15 minutes

---

3. Migration to PostgreSQL (1 hour)

- Oracle / SQL Server object conversion
- Data type mapping
- Sequences & identity columns
- PL/SQL → PL/pgSQL rewrite
- Tools: ora2pg, pgloader, pgAdmin migration

---

4. Post Test + FAQ Session (45 minutes)

A. 40-Question Post Test

- Architecture & MVCC
- Transactions & locking
- Vacuum & bloat
- Query plans & indexing
- Backup & replication concepts
- Migration scenarios

(MCQ + scenario-based questions)

B. Final FAQ & Open Discussion

Typical Questions Covered

- When should Fidelity keep Oracle vs move to PostgreSQL?
- What PostgreSQL issues cause most production outages?
- How does RDS change DBA skill requirements?
- How to prepare PostgreSQL for regulatory audits?

Final Outcome for Fidelity DBA Audience

After this training, participants will be able to:

✓ Support PostgreSQL confidently in **investment & retirement platforms**

✓ Troubleshoot **performance, locking, vacuum, and planner issues**

✓ Understand **RDS PostgreSQL internals without OS access**

✓ Make informed **migration and architecture decisions**

✓ Communicate effectively with **global application teams**